

# **EXHIBIT 12**

Title: Neato Tech Books (J. Blustein)  
 Link: <https://web.cs.dal.ca/~jamie/UWO/Refs/tech-books.html>  
 Captured: Friday, March 12, 2021 15:55:19

## Scholarly Resources for CompSci Undergrads

### Some Useful Technical Books

This list is intended primarily for undergraduate students in computer science at the University of Western Ontario. It is part of my [list of undergrad resources](#) and an experiment with different ways of presenting information.

#### Annotated Version

The links here lead to [full citation information](#).

\*\*\* [Dynamic version](#) of this list!

#### What's Here

1. [Programming in General](#)
2. [Resources About the C Programming Language](#)
3. [Books of and About Algorithms](#)
4. [Other](#)
  - [See Also](#)

#### Programming in General

These are some books about computer programming that I think everyone should read.

##### *The Elements of Programming Style*

The best book about programming and programming style I have ever seen. The examples are in `BATFOR` and `PL/1` but they are short and easy to understand even if you don't know those languages.

##### *Programming Pearls*; and *More Programming Pearls*

Selected essays by Jon Bentley about programming, insights, problems and solutions from the former *Programming Pearls* column in *Communications of the ACM*.

#### Resources about the C Programming Language

##### NOTE

I've been told that the GNU C compiler (`gcc`) that we all know and love (well, okay I like it -- a lot) does not comply with the ANSI standard unless you include *both* the `-ansi` and the `-pedantic` command line switches. Especially if you are learning C just now, make sure you use both of those switches -- and throw in `-Wall` -- for good measure too. I have some more details in my [Notes about gcc](#).

*The C Programming Language (Second edition)* by Kernighan and Ritchie

Not only a great book about both ANSI and K&R C, but a great book about programming (in C). Some example from Kernighan and Plauger's *Software Tools* appear in C form too. You might be able to learn C from this book as easily as any other, but be sure to add your own [cross-references](#), and check both the [errata list](#) and [Koenig's book](#).

If you want to learn C, then you might find my [Advice for Those Who Want to Learn C](#) helpful.

##### *C: A Reference Manual (Fifth Edition)* by Harbison and Steele

The indispensable reference for ANSI C and K&R C. It even includes details about the libraries.

##### The `comp.lang.c` Frequently Asked Questions List

Steve Summit's list is an incredible resource for both new and experienced programmers. An expanded version is available as a book with title *C Programming FAQs: Frequently Asked Questions*.

##### *C Traps and Pitfalls* by Koenig

From the back cover:

Even C experts come across problems that require days of debugging to fix. This book helps to prevent such problems by showing how C programmers get themselves into trouble. Each of the book's many examples has trapped a professional programmer.

#### More? You want more?!

- I think the [C archive](#) in Sweden and its list of related sites are good places to look for more information online.
- Mitch Wright maintains an annotated bibliography of C and Unix books; it is available for anonymous FTP from <ftp://rahul.net> in the `pub/mitch/YABL/` directory.
- I've made an [index](#) of all C related documents by me.

#### Books of and about Algorithms

Where do I go when I can't find it in [Knuth](#) (or don't understand him)? Sometimes I check *Programming Pearls* ([see above](#)) for inspiration and sometimes I start with these books:

Title: Neato Tech Books (J. Blustein)  
 Link: <https://web.cs.dal.ca/~jamie/UWO/Refs/tech-books.html>  
 Captured: Friday, March 12, 2021 15:55:19

Aho, Hopcroft and Ullman's *Data Structures and Algorithms* and *The Design and Analysis of Computer Algorithms*

Although a bit old they still covers some basic algorithms and concepts clearly and thoroughly.

*Introduction to Algorithms* by Cormen *et al.*

The *big white book* covers a lot of territory and covers it well. The presentation of algorithms is helped by the excellent book design.

I think P. J. Plauger's collected *Programming on Purpose* articles will be of great interest to anyone who either does, or intends to, program professionally.

*Programming on Purpose: Essays on Software Design* by P. J. Plauger

Reviewed in ACM Computing Reviews (CR 9311-0825) and IEEE Software (March 1994, pp. 104 - 106)

*Programming on Purpose II: Essays on Software People* by P. J. Plauger

Reviewed in ACM Computing Reviews (CR 9407-0435)

*Programming on Purpose III: Essays on Software Technology* by P. J. Plauger

Reviewed in ACM Computing Reviews (CR 9503-0156)

#### Other

The [finding stuff section](#) of my bookmarks list should help you find all the other references you are looking for. There is also a brief [source code section](#) in that list.

#### SeeAlso

Prof. [Bob Webber](#) created a suggested reading list for anyone who wants to earn a post-graduate degree in Computer Sciences. The readings have been carefully organized over a five year span (12 books/year). I have a copy of the [19 May 1995 version](#) (which was still current in January 1996).

My recommendations for books about writing are [elsewhere](#). I also have a [list of literary books](#) I greatly enjoy.

<http://www.cs.dal.ca/~jamie/UWO/Refs/tech-books.html>

J. Blustein (<http://www.cs.dal.ca/~jamie>)

This document is [copyright](#) by its author, J. Blustein.

